Freeform Search

Database:
- US Pre-Grant Publication Full-Text Database
- **US Patents Full-Text Database**
- US OCR Full-Text Database
- EPO Abstracts Database
- JPO Abstracts Database
- Derwent World Patents Index
- IBM Technical Disclosure Bulletins

Term: `L13 and (return adj address)`

Display: 50 **Documents in Display Format:** REV **Starting with Number** 1

Generate: ○ **Hit List** ● **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Monday, September 25, 2006 Purge Queries Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=USPT; PLUR=NO; OP=OR		
L14	L13 and (return adj address)	27	L14
L13	L12 and array	112	L13
L12	L10 and index	116	L12
L11	L10 and unbound	0	L11
L10	L9 and bits	126	L10
L9	L8 and (transfer or transferring)	126	L9
L8	L7 and cache	147	L8
L7	L6 and (hit or miss)	147	L7
L6	L5 and predict	170	L6
L5	L4 and recovery	496	L5
L4	L3 and (instruction and address and block and stack and fetch)	1776	L4
L3	L2 and 7$$$/$$$.ccls.	20480	L3
L2	(call or subroutine or function) same return	60493	L2
L1	5038281.pn. or 5574873.pn. or 5623617.pn. or 5758140.pn. or 5847954.pn. or 6049866.pn. or 6470435.pn. or 6519768.pn.	8	L1

END OF SEARCH HISTORY

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Monday, September 25, 2006

Hide?	Set Name	Query	Hit Count
		DB=USPT; PLUR=NO; OP=OR	
□	L18	L14 NOT l16	21
□	L17	L16 NOT l14	17
□	L16	L15 and L13	23
□	L15	717/159,141,154.ccls. OR 711/123,141,154,200,202,213,221.ccls. OR 712/200,207,227.ccls. OR 703/26-27.ccls.	6895
□	L14	L13 and (return adj address)	27
□	L13	L12 and array	112
□	L12	L10 and index	116
□	L11	L10 and unbound	0
□	L10	L9 and bits	126
□	L9	L8 and (transfer or transferring)	126
□	L8	L7 and cache	147
□	L7	L6 and (hit or miss)	147
□	L6	L5 and predict	170
□	L5	L4 and recovery	496
□	L4	L3 and (instruction and address and block and stack and fetch)	1776
□	L3	L2 and 7$$$/$$$.ccls.	20480
□	L2	(call or subroutine or function) same return	60493
□	L1	5038281.pn. or 5574873.pn. or 5623617.pn. or 5758140.pn. or 5847954.pn. or 6049866.pn. or 6470435.pn. or 6519768.pn.	8

END OF SEARCH HISTORY

IEEE Xplore® RELEASE 2.1

Home | Login | Logout | Access Information | Alerts |

Welcome United States Patent and Trademark Office

□ Search Results

BROWSE SEARCH IEEE XPLORE GUIDE

Results for "**((((prediction<in>metadata) <and> (return<in>metadata)))<and>(subrouti...**"
Your search matched **2** of **276** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

e-mail

» Search Options

View Session History

New Search

Modify Search

((((prediction<in>metadata) <and> (return<in>metadata)))<and>(subroutine<in>m Search

□ Check to search only within this results set

Display Format: ● Citation ○ Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

view selected items **Select All** **Deselect All**

□ 1. **Microarchitecture support for reducing branch penalty in a superscaler p**
Sakamoto, M.; Nunomura, Y.; Yoshida, T.; Shimazu, Y.;
Computer Design: VLSI in Computers and Processors, 1996. ICCD '96. Procee
IEEE International Conference on
7-9 Oct. 1996 Page(s):208 - 216
Digital Object Identifier 10.1109/ICCD.1996.563559
AbstractPlus | Full Text: PDF(776 KB) **IEEE CNF**
Rights and Permissions

□ 2. **Branch history table prediction of moving target branches due to subrou**
Kaeli, D.R.; Emma, P.G.;
Computer Architecture, 1991. The 18th Annual International Symposium on
May 27-30, 1991 Page(s):34 - 42
AbstractPlus | Full Text: PDF(486 KB) **IEEE CNF**
Rights and Permissions

Help Contact Us Privacy & :

© Copyright 2006 IEEE –

indexed by Inspec®

PORTAL

USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

predict return SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **predict** **return**

Found **54,073** of **185,178**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale □ ▭ ▬ ▬ ■

1 Improving prediction for procedure returns with return-address-stack repair mechanisms ■

Kevin Skadron, Pritpal S. Ahuja, Margaret Martonosi, Douglas W. Clark
November 1998 **Proceedings of the 31st annual ACM/IEEE international symposium on Microarchitecture**
Publisher: IEEE Computer Society Press
Full text available: pdf(1.66 MB) Additional Information: full citation, references, citings, index terms

2 Branch history table prediction of moving target branches due to subroutine returns ■

David R. Kaeli, Philip G. Emma
April 1991 **ACM SIGARCH Computer Architecture News , Proceedings of the 18th annual international symposium on Computer architecture ISCA '91**, Volume 19 Issue 3
Publisher: ACM Press
Full text available: pdf(668.51 KB) Additional Information: full citation, references, citings, index terms

3 Fast and accurate instruction fetch and branch prediction ■

B. Calder, D. Grunwald
April 1994 **ACM SIGARCH Computer Architecture News , Proceedings of the 21ST annual international symposium on Computer architecture ISCA '94**, Volume 22 Issue 2
Publisher: IEEE Computer Society Press, ACM Press
Full text available: pdf(1.07 MB) Additional Information: full citation, abstract, references, citings, index terms

Accurate branch prediction is critical to performance; mispredicted branches mean that ten's of cycles may be wasted in superscalar architectures. Architectures combining very effective branch prediction mechanisms coupled with modified branch target buffers (BTB's) have been proposed for wide-issue processors. These mechanisms require considerable processor resources. Concurrently, the larger address space of 64-bit architectures introduce new obstacles and opportunities. A larger address space ...

4 Case-based reasoning: Predicting outcomes of case based legal arguments ■

Stefanie Bruninghaus, Kevin D. Ashley
June 2003 **Proceedings of the 9th international conference on Artificial intelligence**

and law
Publisher: ACM Press
Full text available: pdf(338.00 KB) Additional Information: full citation, abstract, references

In this paper, we introduce IBP, an algorithm that combines reasoning with an abstract domain model and case-based reasoning techniques to predict the outcome of case-based legal arguments. Unlike the predictions generated by statistical or machine-learning techniques, IBP's predictions are accompanied by explanations.We describe an empirical evaluation of IBP, in which we compare our algorithm to prediction based on Hypo's and CATO's relevance criteria, and to a number of widely used machine le ...

5 Path-based next trace prediction
Quinn Jacobson, Eric Rotenberg, James E. Smith
December 1997 **Proceedings of the 30th annual ACM/IEEE international symposium on Microarchitecture**
Publisher: IEEE Computer Society
Full text available: pdf(1.15 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

The trace cache has been proposed as a mechanism for providing increased fetch bandwidth by allowing the processor to fetch across multiple branches in a single cycle. But to date predicting multiple branches per cycle has meant paying a penalty in prediction accuracy. We propose a next trace predictor that treats the traces as basic units and explicitly predicts sequences of traces. The predictor collects histories of trace sequences (paths) and makes predictions based on these histories. The b ...

Keywords: Trace Cache, Next Trace Prediction, Multiple Branch Prediction, Return History Stack, Path-Based Prediction

6 Applications: Repairing return address stack for buffer overflow protection
Yong-Joon Park, Gyungho Lee
April 2004 **Proceedings of the 1st conference on Computing frontiers**
Publisher: ACM Press
Full text available: pdf(197.90 KB) Additional Information: full citation, abstract, references, index terms

Although many defense mechanisms against buffer overflow attacks have been proposed, buffer overflow vulnerability in software is still one of the most prevalent vulnerabilities exploited. This paper proposes a micro-architecture based defense mechanism against buffer overflow attacks. As buffer overflow attack leads to a compromised return address, our approach is to provide a software transparent micro-architectural support for return address integrity checking. By keeping an uncompromised cop ...

Keywords: buffer overflow, computer architecture, computer security, intrusion tolerance

7 Next cache line and set prediction
Brad Calder, Dirk Grunwald
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture ISCA '95**, Volume 23 Issue 2
Publisher: ACM Press
Full text available: pdf(1.25 MB) Additional Information: full citation, abstract, references, citings, index terms

Accurate instruction fetch and branch prediction is increasingly important on today's wide-issue architectures. Fetch prediction is the process of determining the next instruction to request from the memory subsystem. Branch prediction is the process of predicting the likely out-come of branch instructions. Several researchers have proposed very effective fetch and branch prediction mechanisms including branch target buffers (BTB) that store the target addresses of taken branches. An alternative ...

8 Control Flow Optimization Via Dynamic Reconvergence Prediction
Jamison D. Collins, Dean M. Tullsen, Hong Wang
December 2004 **Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture MICRO 37**
Publisher: IEEE Computer Society
Full text available: pdf(281.24 KB) Additional Information: full citation, abstract, citings

This paper presents a novel microarchitecture technique for accurately predicting control flow reconvergence dynamically. A reconvergence point is the earliest dynamic instruction in the program where we can expect program paths to reconverge regardless of the outcome or target of the current branch. Thus, even if the immediate control flow after a branch is uncertain, execution following the reconvergence point is certain. This paper proposes a novel hardware re-convergence predictor which is b ...

9 Static correlated branch prediction
Cliff Young, Michael D. Smith
September 1999 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 21 Issue 5
Publisher: ACM Press
Full text available: pdf(508.49 KB) Additional Information: full citation, abstract, references, citings, index terms

Recent work in history-based branch prediction uses novel hardware structures to capture branch correlation and increase branch prediction accuracy. Branch correlation occurs when the outcome of a conditional branch can be accurately predicted by observing the outcomes of previously executed branches in the dynamic instruction stream. In this article, we show how to instrument a program so that it is practical to collect run-time statistics that indicate where branch correl ...

Keywords: branch correlation, branch prediction, path profiling, profile-driven optimization

10 Special issue on wireless extensions to the internet: Prediction-based monitoring in sensor networks: taking lessons from MPEG
Samir Goel, Tomasz Imielinski
October 2001 **ACM SIGCOMM Computer Communication Review**, Volume 31 Issue 5
Publisher: ACM Press
Full text available: pdf(1.62 MB) Additional Information: full citation, abstract, references

In this paper we discuss the problem of monitoring data sensed in large sensor networks. A sensor typically runs on a battery having a limited lifetime. In order to increase the lifetime of a sensor it is important that the mechanisms used in monitoring them be energy-efficient. In this paper, we propose a new paradigm called Prediction-based monitoring for energy-efficient monitoring. We show that the paradigm can be visualized as a watching of a "sensor movie" and that concepts from MPEG ma ...

11 Address-Value Delta (AVD) Prediction: Increasing the Effectiveness of Runahead Execution by Exploiting Regular Memory Allocation Patterns
Onur Mutlu, Hyesoon Kim, Yale N. Patt

November 2005 **Proceedings of the 38th annual IEEE/ACM International Symposium on Microarchitecture MICRO 38**
Publisher: IEEE Computer Society
Full text available: pdf(395.75 KB) Publisher Site Additional Information: full citation, abstract

While runahead execution is effective at parallelizing independent long-latency cache misses, it is unable to parallelize dependent long-latency cache misses. To overcome this limitation, this paper proposes a novel technique, address-value delta (AVD) prediction. An AVD predictor keeps track of the address (pointer) load instructions for which the arithmetic difference (i.e., delta) between the effective address and the data value is stable. If such a load instruction incurs a long-latency cach ...

12 Branch prediction for free
Thomas Ball, James R. Larus
June 1993 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1993 conference on Programming language design and implementation PLDI '93**, Volume 28 Issue 6
Publisher: ACM Press
Full text available: pdf(1.49 MB) Additional Information: full citation, abstract, references, citings, index terms

Many compilers rely on branch prediction to improve program performance by identifying frequently executed regions and by aiding in scheduling instructions.Profile-based predictors require a time-consuming and inconvenient compile-profile-compile cycle in order to make predictions. We present a program-based branch predictor that performs well for a large and diverse set of programs written in C and Fortran. In addition to using natural loop analysis to pre ...

13 Meta-heuristics and local search: Evolutionary rule-based system for IPO underpricing prediction
David Quintana, Cristóbal Luque, Pedro Isasi
June 2005 **Proceedings of the 2005 conference on Genetic and evolutionary computation GECCO '05**
Publisher: ACM Press
Full text available: pdf(306.64 KB) Additional Information: full citation, abstract, references, index terms

Academic literature has documented for a long time the existence of important price gains in the first trading day of initial public offerings (IPOs).Most of the empirical analysis that has been carried out to date to explain underpricing through the offering structure is based on multiple linear regression. The alternative that we suggest is a rule-based system defined by a genetic algorithm using a Michigan approach. The system offers significant advantages in two areas, 1) a higher predictive ...

Keywords: genetic algorithm, initial public offering, underpricing

14 Two-level adaptive training branch prediction
Tse-Yu Yeh, Yale N. Patt
September 1991 **Proceedings of the 24th annual international symposium on Microarchitecture**
Publisher: ACM Press
Full text available: pdf(1.13 MB) Additional Information: full citation, references, citings, index terms

15 Control flow prediction for dynamic ILP processors

Dionisios N. Pnevmatikatos, Manoj Franklin, Gurindar S. Sohi
December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**
Publisher: IEEE Computer Society Press
Full text available: pdf(1.44 MB) Additional Information: full citation, references, citings

16 CAVA: Using checkpoint-assisted value prediction to hide L2 misses
Luis Ceze, Karin Strauss, James Tuck, Josep Torrellas, Jose Renau
June 2006 **ACM Transactions on Architecture and Code Optimization (TACO)**, Volume 3 Issue 2
Publisher: ACM Press
Full text available: pdf(646.62 KB) Additional Information: full citation, abstract, references, index terms

Modern superscalar processors often suffer long stalls because of load misses in on-chip L2 caches. To address this problem, we propose hiding L2 misses with Checkpoint-Assisted VAlue prediction (CAVA). On an L2 cache miss, a predicted value is returned to the processor. When the missing load finally reaches the head of the ROB, the processor checkpoints its state, retires the load, and speculatively uses the predicted value and continues execution. When the value in memory arrives at the L2 cac ...

Keywords: Value prediction, checkpointed processor architectures, memory hierarchies, multiprocessor

17 Corpus-based static branch prediction
Brad Calder, Dirk Grunwald, Donald Lindsay, James Martin, Michael Mozer, Benjamin Zorn
June 1995 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1995 conference on Programming language design and implementation PLDI '95**, Volume 30 Issue 6
Publisher: ACM Press
Full text available: pdf(1.35 MB) Additional Information: full citation, abstract, references, citings, index terms

Correctly predicting the direction that branches will take is increasingly important in today's wide-issue computer architectures. The name program-based branch prediction is given to static branch prediction techniques that base their prediction on a program's structure. In this paper, we investigate a new approach to program-based branch prediction that uses a body of existing programs to predict the branch behavior in a new program. We call this approach to program-based ...

18 Increasing the instruction fetch rate via multiple branch prediction and a branch address cache
Tse-Yu Yeh, Deborah T. Marr, Yale N. Patt
August 1993 **Proceedings of the 7th international conference on Supercomputing**
Publisher: ACM Press
Full text available: pdf(1.13 MB) Additional Information: full citation, references, citings, index terms, review

19 From devices to tasks: automatic task prediction for personalized appliance control
Charles L. Isbell, Olufisayo Omojokun, Jeffrey S. Pierce
July 2004 **Personal and Ubiquitous Computing**, Volume 8 Issue 3-4
Publisher: Springer-Verlag
Full text available: pdf(489.15 KB) Additional Information: full citation, abstract, index terms

One of the driving applications of ubiquitous computing is *universal appliance interaction*: the ability to use arbitrary mobile devices to interact with arbitrary appliances, such as TVs, printers, and lights. Because of limited screen real estate and the plethora of devices and commands available to the user, a central problem in achieving this vision is predicting which appliances and devices the user wishes to use next in order to make interfaces for those devices available. We believe ...

Keywords: Appliance, Machine learning, Ubiquitous computing, User interaction

20 Accurate indirect branch prediction

Karel Driesen, Urs Hölzle

April 1998 **ACM SIGARCH Computer Architecture News , Proceedings of the 25th annual international symposium on Computer architecture ISCA '98**, Volume 26 Issue 3

Publisher: IEEE Computer Society, ACM Press

Full text available: pdf(1.49 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Indirect branch prediction is likely to become increasingly important in the future because indirect branches occur more frequently in object-oriented programs. With misprediction rates of around 25% on current processors, indirect branches can incur a significant fraction of branch misprediction overhead even though they remain less frequent than the more predictable conditional branches. We investigate a wide range of two-level predictors dedicated exclusively to indirect branches. Starting wi ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2006 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player